



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/764,745	01/26/2004	Beat Stamm	14984.33	8753
47973	7590	07/15/2005	EXAMINER	
WORKMAN NYDEGGER/MICROSOFT 1000 EAGLE GATE TOWER 60 EAST SOUTH TEMPLE SALT LAKE CITY, UT 84111			WOODS, ERIC V	
			ART UNIT	PAPER NUMBER
			2672	

DATE MAILED: 07/15/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/764,745	STAMM ET AL.
	Examiner	Art Unit
	Eric V. Woods	2672

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 26 January 2004.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-20 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-20 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on 26 January 2004 is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s)/Mail Date: _____
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)
3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date <u>20040126</u>	6) <input type="checkbox"/> Other: _____

DETAILED ACTION

Drawings

Examiner accepts the drawings.

Specification

Applicant is reminded of the duty to disclose under 37 CFR 1.56, and as part of that, MPEP 2001.06(b), third paragraph, clearly requires applicant to disclose all relevant and/or timely filed copending applications. Such applications must be noted in the specification, and applicant must amend the specification to include such applications that examiner has found, with specific ones set forth below. Specifically, applicant filed 10/764961, 10/764787, and 10/764622, which all appear to relevant subject matter, on the same day as the instant application.

Further, applicant is required to examine all co-pending applications filed by the present inventive entities, and determine which ones are relevant and disclose similar subject matter. Examiner **must** know these applications in order to make double patenting determinations, and applicant is expected to disclose such (see MPEP 2004).

The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

Claim Objections

Claim 17 is objected to because of the following informalities: there is no spacing in the term "claim1" which should clearly read "claim 1". Appropriate correction is required.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter, which the applicant regards as his invention.

Claim 7 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, the language in the third line of the claim renders the metes and bounds of the claim unclear. Examiner believes that the line should read, "accessing a parameter that represents how the glyphs of the scaled font are to be offset wherein ...". Otherwise, the claim does not make sense.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.

Art Unit: 2672

2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1, 19, and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rappoport et al (WO 98/36630)('Rappoport').

Claims 19 and 20 are a system and computer program product implementing the method of claim 1. Clearly, software implementing a method that clearly is intended to be computer-implemented is subject to the same rejection without further comment. The only difference between claim 19 and the method claim is that it requires a computer with a processor and memory; *prima facie*, a general-purpose digital computer must inherently contain those components (see for example the works by Turing and Von Neumann to this effect from the 1940s and 1950s). Therefore, since the references applied teach a computer-implemented method that would be inherently taught by those references. Finally, it would be obvious that a software program for making a computer execute a set of instructions is very clearly running on such a digital computer. Thusly, the limitation of "one or more processors" is met. Essentially claim 20 merely recites a computer executing the program of claim 19.

The In response to applicant's arguments, the recitations in the various preambles of claims 19 and 20 that are different from that of claim 1 have not been given patentable weight because the recitation occurs in the preamble. A preamble is generally not accorded any patentable weight where it merely recites the purpose of a process or the intended use of a structure, and where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural

limitations are able to stand alone. See *In re Hirao*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951).

As to claims 1, 19, and 20,

In a computing system that has access to one or more fonts, each font including glyphs representing the corresponding characters of the font, a method for using externally parameterizeable constraints to synthesize a font variant, the method comprising: (Rapoport clearly teaches that the system of LiveType is a dynamic font model (pages 4-5), and in the Background it is disclosed that computers have multiple fonts (pages 1-3). Further, the use of external font parameters is taught in the Abstract, where it clearly states that the behavior of letters is altered by the manipulation of external parameters.

In figure 1, it is shown that the font consists of glyph model 10 (see pages 10-13), where the glyph model contains various elements, the glyph geometry per se – element 12 – and the feature hierarchy 14. The feature hierarchy contains elements of each character, e.g. bars, stems, and serifs. Further, Rapoport clearly teaches that the constraints 22 (page 10) are mapped to parameters, and that such constraints are generated from the attributed parameters 24, which are directly linked to external parameters. As discussed at the bottom of claim 24, a constraint specifying the distance between two points could be “d”, where ‘d’ can be used as a parameter per se. Clearly, after applying all of the limitations to the font, the scaled character would be a font variant per se. Further, page 12, line 20 – page 13, line 25, where especially in page 12, lines 21-28, it is stated that parameters are the external interface to dynamic behaviors of the features and glyphs. Further, on page 14, lines 14-30 and Figures 6A-

Art Unit: 2672

6D and 7A-7B clearly show how the user can change external parameters, namely the 'Beta Round' parameter and the effect this will have on exemplary characters. Further, external parameters can be used to change constraints – e.g. 14:13-18.)

-Accessing a scaled font that has been scaled for rendering at a target size and a target resolution, the scaled font referencing hints that constrain how glyphs of the scaled font are to be rendered at the target size and target resolution; (Rapoport clearly teaches on page 3 that, "High quality rasterization of outline fonts involves grid fitting constraints called 'hints'." Very clearly, the constraints used by Rapoport are hints as the term is defined in the art. Further, on page 5 in the summary of the invention, it clearly states on page 5, lines 3-7, that a glyph is defined by its geometry, and that on lines 20-22 that fonts are defined by boundary and support points, which are comparable to the control points that define standard fonts in 3:3-10. The version of Rapoport is a more advanced version of this technology. Also, to illustrate this point further, 17:5-11 teaches that Rapoport uses grid-scaling methods that are used by prior art fonts that use the 'hints' recited in the instant claim. Further, the system of Rapoport has a global font model (23:19-24:25), which clearly minimizes memory space required to store the font. Secondly, the global model is invoked whenever a character is required to be displayed. In 24:20 – 25:20 the process of rendering is described; the final affine transformations are computed, that is the font is computed for display purposes, and then the glyph outline is computed. Then the outline is rasterized, and a final bitmap is produced, while the rasterizing hints are used during this process. Clearly, the global font model and separate glyph geometries constitute a font. A font must prima facie be

displayed on a computer monitor for it to be useful; in order for a typographer to be able to examine the font, it must be shown on the screen so that the results of the alterations can be viewed. As such, a font will be stored as it is generated. A display device has – at any given moment – a given resolution that it is operative at (for example, under Microsoft® Windows™, right-clicking on the desktop and clicking the Properties option brings up a window allowing the user to set the resolution of the screen). The point of this exercise is that when a font glyph / character is displayed, the final, scaled version will be generated for a target resolution – whatever the instant, operating resolution of the display device – and it will have a given size that is directly proportional to the resolution (16:26-17:5), since all languages line characters on lines, and the distance between lines is dependent upon the size of the display device and the set spacing, which will (at the instant the character is finally generated for display on the screen) be known. As such, a display that is 17" wide running at SVGA (1024 x 768) resolution showing characters in a word processing program that has single spaced lines and a given size (e.g. 12-point) will have a specific, set size for the characters – a height of n pixels, where n is the approximate height of a line minus whatever guard bands / blank pixel spaces kept blank to avoid character overlap and show the appearance of single space. As such, characters are also known to have specific sizes in the x and y directions – see 17:5-17:15. Therefore, a font under the Rappoport model will inherently be 'scaled' to whatever resolution and size the application, operating system, and display device support as set forth above.)

Art Unit: 2672

-Accessing one or more external font parameters that alter how the glyphs of the scaled font are to be rendered; and (As stated in the response to the preamble, Rappoport clearly teaches changing external font parameters in the Abstract and in the locations stated above. Further, in 15:1-10 and in other locations, a user interface to simultaneously adjust several external font parameters is taught, and these clearly effect how the font is rendered, for example, one is listed in Figs. 8A-8D, and for example the width of a stem is changed in Figs. 4A-4B, as noted in the first paragraph. See the discussion for the other clauses of this claim for more detail.)

-Applying the one or more external font parameters to the scaled font to synthesize a font variant such that the hints from the scaled font are preserved in the font variant. (Font variants are clearly computed in Rappoport 12:25-30, where it is stated that modifying a parametric value in a certain feature will cause the constraints connected with the feature to become invalid and the constraints to be reevaluated. As such, a new glyph variation will be created, which very clearly constitutes a 'font variant' given, that as in 14:15-14:30, where it is taught that certain parameters that control over all characters in a font (e.g. global control over shapes using a single parameter (14:13-17 and Figs. 6A-6C for example)) can be changed; that is, one parameter can be adjusted that causes global changes in a font. Very clearly, as taught in 12:25-30, if a global parameter is varied, all glyphs will have their constraints invalidated and new versions of all glyphs (e.g. a new font variant) will be created. Clearly, once a change to a parameter is made, it is applying to the scaled font, or the original base font.)

As stated above, Rappoport clearly teaches all the limitations all of the instant claim, and it would have been obvious to modify Rappoport in any minor fashion necessary to meet the instant claim if necessary (examiner asserts that Rappoport indeed teaches all the limitations of the instant claim as written above).

As to claim 2,

The method as recited in claim 1, wherein accessing a scaled font that has been scaled for rendering at a target size and a target resolution comprises accessing a scaled font that includes font-hinting language instructions.

Rappoport clearly teaches as in claim 1 that hints are used in rendering fonts – see 25:10-11, where rasterization hints are still used. Further, as stated above in the rejection to claim 1, Rappoport teaches a new paradigm where each glyph has a geometry associated with it, and various features that consist of boundary point and support points, that are modified in keeping with known constraints and those constraints are altered by and controlled with external parameters. Clearly, these constraints are comparable to the hints as set forth above. Further, as established in claim 1, grid-based scaling is still used on the boundary points. Rappoport further teaches that the global constraints and system states presented in 23:1-16 allow new programming language paradigms for programming the state-based constraint of the Rappoport invention 4:9-15 and 4:24-5:3. Finally, it is well established by Rappoport that systems like Postscript Type 1 and TrueType (3:3-10) are programming languages that allow typographers to specify constraints for fonts and glyphs, and the metafont system (2:10-22) allows for programming of font hints as well. Obviously, standard

hints from TrueType and Postscript could easily be used with the system of the Rappoport application, since such hints are still used during the rasterizing process. This claim is a trivially obvious variant of claim 1, and motivation for modification is taken from the rejection to the parent claim, and is also provided by the fact that being able to program such constraints makes authoring a font take less time than doing all changes graphically. Clearly, the scaled font will have the hinting language instructions for the reasons set forth above, since they are a fundamental part of rendering the font in the first place.

As to claim 3, Rappoport clearly teaches (as in claim 1) that the user can manipulate font parameters. Further, in 15:1-10 and in other locations, a user interface to simultaneously adjust several external font parameters is taught, and these clearly effect how the font is rendered, for example, one is listed in Figs. 8A-8D, and for example the width of a stem is changed in Figs. 4A-4B, as noted in the first paragraph. Also, in Fig. 5 it is shown how the user can manipulate a character and can change the vertical and horizontal amount of compression (e.g. the position) as recited by applicant in the instant claim (see Rappoport 15:1-25).

As to claim 4, this is a substantial duplicate of claim 3, except that the characters are being expanded rather than compression. For example Figure 5 shows both compression and expansion in at least a horizontal or vertical direction. The rejection to claim 3 is incorporated by reference.

As to claims 5 and 6, Rappoport in 15:1-15 and Figs. 8A-8D teach that the degree of bolding of a character can be changed, wherein bolding clearly changes the

Art Unit: 2672

weight of a character as defined in the instant claim. Clearly, changing the degree of bolding (and in Fig. 5) allows the changes to go in either direction – compression or expansion.

As to claim 7, this is a trivially obvious variation. Offsetting a glyph in the vertical direction comprises generating superscripts or subscripts. It is well known in the art to perform this step – standard word processing software such as Microsoft® Word™ clearly allows the user to make this kind of change, and it would be obvious that the user should be able to control the degree of offset in order to emphasize or control the degree of emphasis of superscript or subscript or footnoting. It would have been trivially obvious to modify Rappoport to allow the user to modify the degree of offset, since it is a well-known part of typography. Examiner also takes Official Notice of this fact.

As to claim 8, this claim is a substantial duplicate of claim 2, the rejection to which is incorporated by reference, where the claimed step is the same. Since this is a method claim written using “comprising” language, it is open ended; since the method claim is implemented using software, it would be obvious that the steps can be arbitrarily rearranged. Finally, since it has been established that TrueType™ and similar prior art font hinting languages allow the user to specify constraints, it would be obvious that since the system of Rappoport allows users to program it, that the external parameters could clearly be programmed into the constraint state machine that is known to be programmable, which would fulfill the limitations of this step.

As to claim 9 and 10, they are substantial duplicates of claims 3 and 4 (the rejections to which are incorporated by reference), in that the recited limitations are the

Art Unit: 2672

same, they are simply applied to different steps in the method. This would be a trivially obvious variant (design choice as to where to place the step in the method and/or computer program, where it has been well established that software can be written in modules and/or infinitely arbitrarily rearranged). Finally, clearly applying the external parameters using the interface specified in claim 1 (for example to effect the bolding, or simply simultaneously changing several parameters as set forth in previous rejections above) would require that the glyphs be redrawn (as stated in the discussion of the last clause of claim 1).

As to claim 11, it is a substantial duplicate of claim 7, the rejection to which is incorporated by reference. By the logic set forth in the rejections for claims 9 and 10 above, the limitations are obvious.

As to claim 12, see Rappoport Figs. 18A-18D and Figs. 19-21, where clearly a common, standardized distance is maintained between glyphs regardless of variations in parameters applied globally or to each glyph separately. In Figs. 18A-18C, the reference heights are maintained. Also, for the reasons discussed earlier, see the rejection to claim 7 – languages are dependent upon line spacing and the rejection to claim 1 sets that forth in more detail – it would be obvious that maintaining a reference height and spacing would be useful and therefore obvious, and the system / method of Rappoport clearly can perform the recited limitation, as demonstrated in the cited Figures.

As to claim 13, this limitation is taught in the rejection to claim 1, where the alteration of a global parameter causes all relevant or effected glyphs to be recalculated

Art Unit: 2672

a new font variant created. As stated therein, the fonts are rendered / rasterized using known hints, and further they are *prima facie* rendered with the changes in the external font parameters as set forth in the rejection to claim 1.

As to claim 14, this is a trivially obvious variant of claim 12, where the limitations from that claim are merely added to claim 13. As such, that rejection is incorporated by reference.

As to claim 15, scan conversion is the method used to specify which pixels to fill and what to fill them with (definition taken from

<http://www.cs.berkeley.edu/~ddgarcia/cs184/r3/>). Clearly, this is another name for and is synonymous with the process of rasterization, which clearly is performed by Rappoport as stated in the rejection to claim 1, where the creation of a new font variant involves rasterizing using the set external hints. Also, it is established in claim 1 that the system of Rappoport uses font outlines (the boundary points for example). Therefore, all the limitations of this claim are met.

As to claim 16, this again corresponds to the process of rasterization, which inherently generates a bit-map of the glyph. Clearly, a bit-map is defined by pixels, where each pixel includes red, green, and blue components (channels) for display on a display device, where the intensity of each of the sub-components of a pixel is set by the intensity value of the RGB channels. Clearly, the utilization of rasterization hints as taught by Rappoport covers this limitation. Further, the entire purpose of the Rappoport invention is to more accurately render glyphs (4:1-15). *Prima facie*, these glyphs will comply with the hints and external constraints, because they are only generated when

changes in the parameters cause the constraints to be invalidated, and the newly generated glyphs will *prima facie* result in valid constraints.

As to claim 17, this step is inherent. That is, in order for a font to be used as part of a document and in order for a font to be used as part of a document (e.g. the Figures of the instant application), the glyphs of a font file (as in element 100 in Rappoport) would *prima facie* be rendered and rasterized (e.g. scaled) according to the capabilities of the display device and the operating system and applications (e.g. line spacing, inherent instant device resolution, et cetera). Examiner further takes Official Notice of the fact.

Claim 18 is rejected as unpatentable and obvious under 35 U.S.C. 103(a) over Rappoport in view of Betrisey et al (US PGPub 2001/00448764 A1).

Rappoport does not explicitly teach this limitation while Betrisey does. The idea of caching anything is well known in the art, because it speeds access to whatever is cached, and it would be obvious to apply it in this case. Betrisey teaches in [0034] the caching of the raster bitmaps (e.g. the rasterized version of a font), which clearly constitutes "caching the font variant" so that it can be more efficiently accessed in response to the request of an application program. Motivation for combination is provided by the above cited fact – that caching speeds access and is inherently more efficient (see also [0035] and [0038]).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric V. Woods whose telephone number is 571-272-7775. The examiner can normally be reached on M-F 7:30-4:30 alternate Fridays off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Michael Razavi can be reached on 571-272-7664. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Eric Woods
July 9, 2005

Jeffery A. Burns
JEFFERY BURNS
PRIMARY EXAMINER